

Amendments to the Specification:

At page 5, please replace the first paragraph as follows:

It will be appreciated that situations may arise in which it is desired to delete a selected Phase 1 SA without deleting related Phase 2 SAs. For example, a system administrator may ~~which~~ wish to manually clear the Phase 1 SA. Alternatively, an error relating to the Phase 1 SA may be detected and, in response, automatic or manual deletion of the Phase 1 SA may be initiated. However, according to the conventional IKE and IPSec protocols, the deletion of any given Phase 1 SA having active, associated Phase 2 SAs will result in the Phase 1 renegotiation. Thus, according to conventional techniques, even if it is desired to delete a selected Phase 1 SA, the existence of any related, active Phase 2 SA will result in the Phase 1 SA continuously renegotiating itself. Accordingly, it will be appreciated that there exists a general desire to improve upon security communication protocols implemented in IP networks.

At pages 9-10, please replace the last paragraph as follows:

FIGURE 3 shows an example of a Control Message Processing Procedure 300 in accordance with a specific embodiment of the present invention. In one embodiment, the procedure 300 may be implemented at a network device which receives the modified control message described, for example, in FIGURE 1 of the drawings. As shown at 302 of FIGURE 3, the Control Message Processing Procedure may be triggered in response to receiving a new control message at the receiving network device. The modified control message is then analyzed (304) in order to extract appropriate information from the control message, including reason information (if any) which may be identified in the reason field of the modified control message. Using the control message information, the network device may then determine (306) and implement (308) an appropriate response to the control message ~~respond (306) to the control message as appropriate~~. As described previously, the appropriate response by the network node may be based, at least in part, upon the information retrieved from the reason field of the modified control message.

At page 11, beginning at line 3, please replace the paragraph with the following paragraph:

CPU 62 may include one or more processors 63 such as a processor from the Motorola family of microprocessors or the MIPS (million instructions per second) family of microprocessors. In an alternative embodiment, processor 63 is specially designed hardware for controlling the operations of network device 60. In a specific embodiment, a memory 61 (such as non-volatile RAM and/or ROM) also forms part of CPU 62. However, there are many different ways in which memory could be coupled to the system. Memory block 61 may be used for a variety of purposes such as, for example, caching and/or storing data, programming instructions, etc.

At page 11, at the paragraph beginning at line 11, please replace the paragraph with the following paragraph:

The interfaces 68 are typically provided as interface cards (sometimes referred to as “line cards”). Generally, they control the sending and receiving of data packets over the network and sometimes support other peripherals used with the network device 60. Among the interfaces that may be provided are Ethernet interfaces, frame relay interfaces, cable interfaces, DSL interfaces, token ring interfaces, and the like. In addition, various very high-speed interfaces may be provided such as fast Ethernet interfaces, Gigabit Ethernet interfaces, ATM interfaces, HSSI (High-Speed Serial Interface) interfaces, POS (Point of Sale) interfaces, FDDI (Fiber Distributed Data Interface) interfaces and the like. Generally, these interfaces may include ports appropriate for communication with the appropriate media. In some cases, they may also include an independent processor and, in some instances, volatile RAM. The independent processors may control such communications intensive tasks as packet switching, media control and management. By providing separate processors for the communications intensive tasks, these interfaces allow the master microprocessor 62 to efficiently perform routing computations, network diagnostics, security functions, etc.